



## UniqTag: Content-derived unique and stable identifiers for gene annotation

Shaun D Jackman, Joerg Bohlmann and Inanc Birol

bioRxiv first posted online August 1, 2014

Access the most recent version at doi: <http://dx.doi.org/10.1101/007583>

---

Creative  
Commons  
License

The copyright holder for this preprint is the author/funder. It is made available under a [CC-BY 4.0 International license](#).

# UniqTag: Content-derived unique and stable identifiers for gene annotation

Shaun Jackman<sup>1,2,\*</sup>, Joerg Bohlmann<sup>3,4</sup> and İnanç Birol<sup>1,5 \*</sup>

<sup>1</sup>Genome Sciences Centre, British Columbia Cancer Agency, Vancouver, BC, Canada

<sup>2</sup>Graduate Program in Bioinformatics, University of British Columbia, Vancouver, BC, Canada

<sup>3</sup>Michael Smith Laboratories, University of British Columbia, Vancouver, BC, Canada

<sup>4</sup>Department of Forest Science, University of British Columbia, Vancouver, BC, Canada

<sup>5</sup>Department of Medical Genetics, University of British Columbia, Vancouver, BC, Canada

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Associate Editor: XXXXXXXX

## ABSTRACT

### Summary

When working on an ongoing genome sequencing and assembly project, it is rather inconvenient when gene identifiers change from one build of the assembly to the next. The gene labelling system described here, UniqTag, addresses this common challenge. UniqTag assigns a unique identifier to each gene that is a representative  $k$ -mer, a string of length  $k$ , selected from the sequence of that gene. Unlike serial numbers, these identifiers are stable between different assemblies and annotations of the same data without requiring that previous annotations be lifted over by sequence alignment. We assign UniqTag identifiers to nine builds of the Ensembl human genome spanning seven years to demonstrate this stability.

### Availability and implementation

The implementation of UniqTag is available at

<https://github.com/sjackman/unihtag>

Supplementary data and code to reproduce it is available at

<https://github.com/sjackman/unihtag-paper>

### Contact

Shaun Jackman <[sjackman@bcgsc.ca](mailto:sjackman@bcgsc.ca)>

Inanc Birol <[ibirol@bcgsc.ca](mailto:ibirol@bcgsc.ca)>

## 1 INTRODUCTION

The task of annotating the genes of a genome sequence often follows genome assembly. These annotated genes are assigned unique identifiers by which they can be referenced. Assembly and annotation is frequently an iterative process, by refining the method or by the addition of more sequencing data. These gene identifiers would ideally be stable from one assembly and annotation to the next. The common practice is to use serial numbers to identify genes that are annotated by software such as MAKER (Campbell, 2014), which, although certainly unique, are not stable between assemblies. A single change in the assembly can result in a total renumbering of the annotated genes.

One solution to stabilize identifiers is to assign them based on the content of the gene sequence. A cryptographic hash function such

as SHA (Secure Hash Algorithm) (Dang, 2012) derives a message digest from the sequence, such that two sequences with the same content will have the same message digest, and two sequences that differ will have different message digests. If a cryptographic hash were used to identify a gene, the same gene in two assemblies with identical content would be assigned identical identifiers. Yet, by design a slight change in the sequence, such as a single-character substitution, would result in a completely different digest.

Locality-sensitive hashing in contrast aims to assign items that are similar to the same hash value. A hash function that assigns an identical identifier to a sequence after a modification of that sequence is desirable for labelling the genes of an ongoing genome annotation project. One such locality-sensitive hash function, MinHash, was employed to identify web pages with similar content (Broder, 1997) by selecting a small representative set of words from a web page.

UniqTag is inspired by MinHash. It selects a single representative  $k$ -mer from a sequence to assign a stable identifier to a gene. These identifiers are intended for systematic labelling of genes rather than assigning biological gene names, as the latter are typically based on biological function or homology to orthologous genes.

## 2 METHODS

The UniqTag is defined mathematically as follows. Let  $\Sigma$  be an alphabet, such as the twenty standard amino acids or the four nucleotides. Let  $\Sigma^k$  be the set of all strings over  $\Sigma$  of length  $k$ . Let  $s$  be a string over  $\Sigma$ , such as the peptide or nucleotide sequence of a gene. Let  $C(s)$  be the set of all substrings of  $s$ , and  $C_k(s)$  be the set of all  $k$ -mers of  $s$ , that is, all substrings of  $s$  with length  $k$ .

$$C_k(s) = C(s) \cap \Sigma^k$$

Let  $S$  be a set of  $n$  strings  $\{s_0, \dots, s_n\}$ , such as the peptide or nucleotide sequences of the annotated genes of a genome assembly. Let  $f(t, S)$  be the frequency in  $S$  of a  $k$ -mer  $t$ , defined as the number of strings in  $S$  that contain the  $k$ -mer  $t$ .

$$f(t, S) = |\{s \mid t \in C_k(s) \wedge s \in S\}|$$

\*to whom correspondence should be addressed

Let  $T$  be the set of  $k$ -mers of  $t$ , and  $\min T$  be the lexicographically minimal  $k$ -mer of  $T$ . If the  $k$ -mers of  $T$  were sorted alphabetically, it would be the first  $k$ -mer in the list.

Finally,  $u_k(s, S)$  is the UniqTag, the lexicographically minimal  $k$ -mer of those  $k$ -mers of  $s$  that are least frequent in  $S$ .

$$u_k(s, S) = \min_{t \in C_k(s)} \arg \min f(t, S)$$

Typically,  $u_k(s, S)$  is the first  $k$ -mer in an alphabetically sorted list of the  $k$ -mers of a gene that are unique to that gene.

A UniqTag can be generated from the nucleotide sequence of a gene or the translated peptide sequence of a protein-coding gene. Using the peptide sequence results in a UniqTag that is stable across synonymous changes to the coding sequence as well as to changes in the untranslated regions and introns of the gene. Since the amino acid alphabet is larger than the nucleotide alphabet, fewer characters are required for a  $k$ -mer to be likely unique, resulting in an aesthetically pleasing shorter identifier.

When two gene models have identical  $k$ -mer compositions, they would be assigned the same UniqTag. It is also possible that two genes that have no unique  $k$ -mer and similar  $k$ -mer composition are assigned the same UniqTag. In such cases, genes that have the same UniqTag are distinguished by adding a numerical suffix to the UniqTag.

The UniqTag is designed to be stable but will change in the following conditions: (1) when the sequence at the locus of the UniqTag changes; (2) when a least-frequent  $k$ -mer that is lexicographically smaller than the previous UniqTag is created; (3) when a duplicate  $k$ -mer is created elsewhere that results in the previous UniqTag no longer being a least-frequent  $k$ -mer.

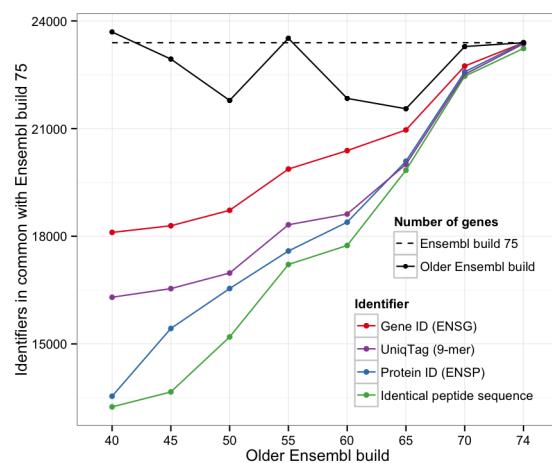
The special cases of merging and splitting gene models are interesting. Concatenating two gene models results in a gene whose UniqTag is the minimum of the two previous UniqTags, unless the new UniqTag spans the junction of the two sequences. Similarly when a gene model is split in two, one gene is assigned a new UniqTag and the other retains the previous UniqTag, unless the previous UniqTag spanned the junction.

Importantly and in contrast, unlike naming the genes after the genomic contigs or scaffolds in which they are found, changing the order of the genes in a genome assembly has no effect on the UniqTag.

### 3 RESULTS

To demonstrate the stability and utility of UniqTag, we assigned identifiers to the genes of nine builds of the Ensembl human genome (Flicek, 2014) spanning seven years and two major genome assemblies, NCBI36 up to build 54 and GRCh37 afterward. An identifier of nine peptides ( $k = 9$ ) was assigned to the first protein sequence, that with the smallest Ensembl protein (ENSP) accession number, of each gene. The number of common UniqTag identifiers between older builds from build 40 on and the current build 75 is shown in Figure 1. Also shown is the number of common gene and protein identifiers (ENSG and ENSP accession numbers) between builds and the number of genes with peptide sequences that are identical between builds. Although less stable than the gene ID, the UniqTag is more stable than the protein ID and the peptide sequence.

Whereas the gene and protein identifiers can, with effort, be lifted over from older builds to the newest build, the UniqTag identifier can be generated without any knowledge of previous assemblies, making it a much simpler operation. The number of identical peptide sequences between builds shows the stability that would be expected of using a cryptographic hash value of the peptide sequence as the identifier. Supplementary Figure S1 shows that the UniqTag stability is insensitive to the size of the UniqTag identifier for values of  $k$  between 8 and 50 peptides. The data for these figures are shown in supplementary Table S1.



**Fig. 1.** The number of common UniqTag identifiers between older builds of the Ensembl human genome and the current build 75, the number of common gene and protein identifiers between builds, and the number of genes with peptide sequences that are identical between builds.

### ACKNOWLEDGEMENTS

The authors thank Nathaniel Street for his enthusiastic feedback, the SMaRTForests project and the organizers of the 2014 Conifer Genome Summit that made our conversation possible.

**Funding:** This work was supported by the Natural Sciences and Engineering Research Council of Canada, Genome British Columbia, Genome Quebec and Genome Canada.

### REFERENCES

- Broder, A. Z. (1997) On the resemblance and containment of documents. *Compression and Complexity of Sequences*, 1997 Proceedings, 21-29.
- Campbell, M. S. et al. (2014) MAKER-P: a tool-kit for the rapid creation, management, and quality control of plant genome annotations. *Plant Physiology*, 164(2), 513-524.
- Dang, Q. H. (2012) Secure Hash Standard (SHS). *NIST FIPS*, 180(4), 1-35.
- Flicek, P. et al. (2014) Ensembl 2014. *Nucleic Acids Research*, 42(D1), D749-D755.